

CLAIMS

What is claimed is:

1 1. A method for deriving transformations for transforming data
2 from one data schema to another, comprising:
3 receiving a source data schema and a target data schema;
4 mapping the source data schema into an ontology model;
5 mapping the target data schema into the ontology model; and
6 deriving a transformation for transforming data conforming to
7 the source data schema into data conforming to the target data schema, using the
8 ontology model.

1 2. The method of claim 1 further comprising converting at least one
2 of the source data schema and the target schema from an external format to an
3 internal format.

1 3. The method of claim 1 further comprising receiving the ontology
2 model.

1 4. The method of claim 3 further comprising converting the
2 ontology model from an external format to an internal format.

1 5. The method of claim 1 further comprising generating the
2 ontology model.

1 6. The method of claim 5 further comprising receiving an initial
2 ontology model, wherein said generating generates the ontology model from the
3 initial ontology model.

1 7. The method of claim 6 further comprising converting the initial
2 ontology model from an external format to an internal format.

1 8. The method of claim 1 further comprising generating executable
2 program code that transforms data conforming to the source data schema into data
3 conforming to the target data schema.

1 9. The method of claim 1 wherein the source data schema is a
2 source table schema describing source data tables, wherein the target data schema
3 is a target table schema describing target data tables, and wherein the source table
4 schema and the target table schema each describes at least one table having
5 columns.

1 10. The method of claim 9 wherein the source table schema is a
2 source relational database schema describing source relational database tables,
3 wherein the target table schema is a target relational database schema describing
4 target relational database tables, and wherein the transformation is an SQL query.

1 11. The method of claim 10 wherein said mapping a source data
2 schema and said mapping a target data schema each comprise:
3 identifying at least one class in the ontology model
4 corresponding to at least one table; and
5 identifying at least one property or composition of properties in
6 the ontology model corresponding to at least one table column.

1 12. The method of claim 11 wherein said deriving comprises:
2 labeling properties of the ontology model with symbols;
3 converting at least one column in the source relational database
4 schema into at least one source symbol;
5 converting at least one column in the target relational database
6 schema into at least one target symbol; and
7 expressing the at least one target symbol in terms of at least one
8 source symbol.

1 13. The method of claim 12 wherein said expressing uses
2 expressions involving composition of properties.

1 14. The method of claim 12 wherein at least one dependency exists
2 among properties in the ontology model, and wherein said deriving further
3 comprises translating the at least one dependency among properties in the
4 ontology model as at least one dependency between target relational database
5 columns and source relational database columns, and wherein said expressing
6 incorporates the at least one dependency between target relational database
7 columns and source relational database columns.

1 15. The method of claim 14 wherein said expressing uses
2 expressions involving arithmetic operations.

1 16. The method of claim 14 wherein said expressing uses
2 expressions involving character string operations.

1 17. The method of claim 10 further comprising applying the query to
2 at least one source relational database table to populate at least one target
3 relational database table.

1 18. The method of claim 17 wherein the at least one source relational
2 database table reside in a single database.

1 19. The method of claim 17 wherein the at least one source relational
2 database table reside in multiple databases.

1 20. The method of claim 1 wherein the source data schema is a
2 source document schema describing source documents, and wherein the target
3 data schema is a target document schema describing target documents.

1 21. The method of claim 20 wherein the source document schema is
2 a source DTD describing source XML documents, wherein the target document
3 schema is a target DTD describing target XML documents, and wherein the
4 source DTD and the target DTD each describes at least one XML element or
5 XML attribute.

1 22. The method of claim 21 wherein the transformation is an
2 XQuery.

1 23. The method of claim 21 wherein the transformation is an XSLT
2 script.

1 24. The method of claim 20 wherein the source document schema is
2 a source XML schema describing source XML documents, wherein the target
3 document schema is a target XML schema describing target XML documents, and

2051101502

4 wherein the source XML schema and the target XML schema each describes at
5 least one XML complexType having at least one XML element or XML attribute.

1 25. The method of claim 24 wherein the transformation is an
2 XQuery.

1 26. The method of claim 24 wherein the transformation is an XSLT
2 script.

1 27. The method of claim 24 wherein said mapping a source data
2 schema and said mapping a target data schema each comprise:
3 identifying at least one class in the ontology model
4 corresponding to at least one XML complexType; and
5 identifying at least one property or composition of properties in
6 the ontology model corresponding to at least one XML element or XML attribute.

1 28. The method of claim 24 wherein said deriving comprises
2 expressing XML elements and XML attributes of the target XML schema in terms
3 of XML elements and XML attributes of the source XML schema.

1 29. The method of claim 28 wherein said expressing is performed
2 recursively through XPath paths.

1 30. The method of claim 27 wherein at least one dependency exists
2 among properties in the ontology model, and wherein said deriving further
3 comprises translating the at least one dependency among properties in the
4 ontology model as at least one dependency between target XML elements and
5 source XML elements.

1 31. The method of claim 26 further comprising applying the XSLT
2 script to at least one source XML document to generate at least one target XML
3 document.

1 32. The method of claim 31 wherein the at least one source XML
2 document reside in a single database.

1 33. The method of claim 31 wherein the at least one source XML
2 document reside in multiple databases.

1 34. A system for deriving transformations for transforming data from
2 one data schema to another, comprising:
3 a schema receiver receiving a source data schema and a target
4 data schema;
5 a mapping processor mapping a data schema into an ontology
6 model; and
7 a transformation processor deriving a transformation for
8 transforming data conforming to the source data schema into data conforming to
9 the target data schema, based on respective source and target mappings generated
10 by said mapping processor for mapping said source data schema and said target
11 data schema into a common ontology model.

1 35. The system of claim 34 further comprising a schema format
2 convertor, converting at least one of the source data schema and the target data
3 schema from an external format to an internal format.

1 36. The system of claim 34 further comprising an ontology receiver
2 receiving the ontology model.

1 37. The system of claim 36 further comprising an ontology format
2 convertor, converting the ontology model from an external format to an internal
3 format.

1 38. The system of claim 34 further comprising an ontology builder
2 generating the ontology model.

1 39. The system of claim 38 further comprising an ontology receiver
2 receiving an initial ontology model, wherein said ontology builder generates the
3 ontology model from the initial ontology model.

1 40. The system of claim 39 further comprising an ontology format
2 convertor, converting the initial ontology model from an external format to an
3 internal format.

1 41. The system of claim 34 further comprising a program code
2 generator generating executable program code that transforms data conforming to
3 the source data schema into data conforming to the target data schema.

1 42. The system of claim 34 wherein the source data schema is a
2 source table schema describing source data tables, wherein the target data schema
3 is a target table schema describing target data tables, and wherein the source table
4 schema and the target table schema each describes at least one data table having
5 columns.

1 43. The system of claim 42 wherein the source table schema is a
2 source relational database schema describing source relational database tables,
3 wherein the target table schema is a target relational database schema describing
4 target database tables, and wherein the transformation is an SQL query.

1 44. The system of claim 43 wherein said mapping processor
2 comprises:
3 a class identifier identifying at least one class in the common
4 ontology model corresponding to at least one table; and
5 a property identifier identifying at least one property or
6 composition of properties in the common ontology model corresponding to at
7 least one table column.

1 45. The system of claim 44 wherein said property identifier presents
2 a user with a choice of at least one property in the common ontology model that
3 may correspond to a given table column.

1 46. The system of claim 45 wherein the choice of at least one
2 property only includes properties having targets that are compatible with a data
3 type of the given table column.

1 47. The system of claim 46 wherein, for a given table column that is
2 a foreign key to a foreign table, the choice of at least one property only includes
3 properties whose target is a class corresponding to the foreign table.

1 48. The system of claim 43 wherein said transformation processor
2 comprises:

3 an ontology labeller labeling properties of the common ontology
4 model with symbols;

5 a column converter converting at least one column in the source
6 relational database schema into at least one source symbol, and converting at least
7 one column in the target relational database schema into at least one target
8 symbol; and

9 a symbol processor expressing the at least one target symbol in
10 terms of at least one source symbol.

1 49. The system of claim 48 wherein said symbol processor uses
2 expressions involving composition of properties.

1 50. The system of claim 48 wherein at least one dependency exists
2 among properties in the ontology model, and wherein said transformation
3 processor further comprises a dependency processor translating the at least one
4 dependency among properties in the ontology model as at least one dependency
5 between target relational database columns and source relational database
6 columns, and wherein said symbol processor incorporates the at least one
7 dependency between target relational database columns and source relational
8 database columns.

1 51. The system of claim 50 wherein said symbol processor uses
2 expressions involving arithmetic operations.

1 52. The system of claim 50 wherein said symbol processor uses
2 expressions involving character string operations.

1 53. The system of claim 43 further comprising:
2 a data receiver receiving at least one source relational database
3 table; and
4 a data processor applying the query to the at least one source
5 relational database table to populate at least one target relational database table.

1 54. The system of claim 53 wherein the at least one source relational
2 database table reside in a single database.

1 55. The system of claim 53 wherein the at least one source relational
2 database table resides in multiple databases.

1 56. The system of claim 34 wherein the source data schema
2 comprises a source document schema describing source documents, and wherein
3 the target data schema comprises a target document schema describing target
4 documents.

1 57. The system of claim 56 wherein the source document schema is a
2 source DTD describing source XML documents, wherein the target document
3 schema is a target DTD describing target XML documents, and wherein the
4 source DTD and the target DTD each describes at least one XML element or
5 XML attribute.

1 58. The system of claim 57 wherein the transformation is an
2 XQuery.

1 59. The system of claim 57 wherein the transformation is an XSLT
2 script.

1 60. The system of claim 56 wherein the source document schema
2 comprises a source XML schema that describes XML source documents, wherein
3 the target document schema comprises a target XML schema that describes XML
4 target documents, and wherein the source XML schema and the target XML
5 schema each comprises at least one XML complexType having at least one XML
6 element or XML attribute.

1 61. The system of claim 60 wherein the transformation is an
2 XQuery.

1 62. The system of claim 60 wherein the transformation is an XSLT
2 script.

1 63. The system of claim 60 wherein said mapping processor
2 comprises:
3 a class identifier identifying at least one class in the ontology
4 model corresponding to at least one XML complexType; and
5 an property identifier identifying at least one property or
6 composition of properties in the ontology model corresponding to at least one
7 XML element or XML attribute.

1 64. The system of claim 60 wherein said transformation processor
2 comprises an XML processor expressing XML elements and XML attributes of
3 said target XML schema in terms of XML elements and XML attributes of said
4 source XML schema.

1 65. The system of claim 64 wherein said XML processor operates
2 recursively through XPath paths.

1 66. The system of claim 64 wherein at least one dependency exists
2 among properties in the ontology model, and wherein said transformation
3 processor further comprises a dependency processor translating the at least one
4 dependency among properties in the ontology model as at least one dependency
5 between target XML elements or attributes, and source XML elements or
6 attributes, and wherein said XML processor incorporates the at least one
7 dependency between target XML elements or attributes, and source XML
8 elements or attributes.

1 67. The system of claim 60 further comprising
2 a data receiver receiving at least one source XML document; and
3 a data processor applying the XSLT script to the at least one
4 source XML document to generate at least one target XML document.

1 68. The system of claim 67 wherein the at least one source XML
2 document reside in a single database.

1 69. The system of claim 67 wherein the at least one source XML
2 document reside in multiple databases.

1 70. A method for building an ontology model into which data
2 schema can be embedded, comprising:
3 receiving at least one data schema; and
4 building an ontology model into which the at least one data
5 schema can be embedded.

1 71. The method of claim 70 further comprising converting at least
2 one of the at least one data schema from an external format to an internal format.

1 72. The method of claim 70 wherein the at least one data schema is
2 at least one table schema describing data tables having columns.

1 73. The method of claim 72 wherein the at least one table schema is
2 at least one relational database schema describing relational database tables.

1 74. The method of claim 73 wherein said building an ontology
2 model comprises:
3 providing an initial ontology model;
4 adding classes to the initial ontology model corresponding to
5 tables described in the at least one relational database schema; and
6 adding properties to the initial ontology model corresponding to
7 columns described in the at least one relational database schema.

1 75. The method of claim 74 wherein the initial ontology model is
2 empty.

1 76. The method of claim 74 wherein the initial ontology model is
2 non-empty.

1 77. The method of claim 76 further comprising converting the initial
2 ontology model from an external format to an internal format.

1 78. The method of claim 74 wherein said adding classes is performed
2 by a computer in conjunction with a user.

1 79. The method of claim 78 wherein said adding classes prompts a
2 user to add a class to the ontology model when there is a table described in the at
3 least one relational database schema that does not correspond to an existing class
4 in the ontology model.

1 80. The method of claim 74 wherein said adding classes is performed
2 automatically by a computer.

1 81. The method of claim 80 wherein said adding classes
2 automatically adds a class to the ontology model when there is a table described in

3 the at least one relational database schema that does not correspond to an existing
4 class in the ontology model.

1 82. The method of claim 74 wherein said adding properties is
2 performed by a computer in conjunction with a user.

1 83. The method of claim 82 wherein said adding properties prompts
2 a user to add a property to the ontology model when there is a table column
3 described in the at least one relational database schema that does not correspond
4 to an existing property or composition of properties in the ontology model.

1 84. The method of claim 74 wherein said adding properties is
2 performed automatically by a computer.

1 85. The method of claim 84 wherein said adding properties
2 automatically adds a property to the ontology model when there is a table column
3 described in the at least one relational database schema that does not correspond
4 to an existing property or composition of properties in the ontology model.

1 86. The method of claim 70 wherein said building an ontology
2 model comprises inferring inheritance relationships between classes in the
3 ontology model based on relationships between tables described in the at least one
4 relational database schema.

1 87. The method of claim 86 wherein a first class in the ontology
2 model is inferred to inherit from a second class in the ontology model when a
3 table corresponding to the first class has a primary key that is a foreign key to a
4 table corresponding to the second class.

1 88. The method of claim 86 wherein said inferring inheritance
2 relationships includes prompting a user to confirm an inferred inheritance
3 relationship.

1 89. The method of claim 70 wherein the at least one data schema is
2 at least one document schema describing documents.

1 90. The method of claim 89 wherein the at least one document
2 schema is an XML schema describing XML documents having at least one XML
3 complexType with at least one XML element or XML attribute.

1 91. The method of claim 90 wherein said building an ontology
2 model comprises:
3 providing an initial ontology model;
4 adding classes to the initial ontology model corresponding to
5 XML complexTypes described in the at least one XML schema; and
6 adding properties to the initial ontology model corresponding to
7 XML elements and XML attributes described in the at least one XML schema.

1 92. The method of claim 91 wherein the initial ontology model is
2 empty.

1 93. The method of claim 92 wherein the initial ontology model is
2 non-empty.

1 94. The method of claim 91 wherein said adding classes is performed
2 by a computer in conjunction with a user.

1 95. The method of claim 94 wherein said adding classes prompts a
2 user to add a class to the ontology model when there is an XML complexType
3 described in the at least one XML schema that does not correspond to an existing
4 class in the ontology model.

1 96. The method of claim 91 wherein said adding classes is performed
2 automatically by a computer.

1 97. The method of claim 96 wherein said adding classes
2 automatically adds a class to the ontology model when there is an XML
3 complexType described in the at least one XML schema that does not correspond
4 to an existing class in the ontology model.

1 98. The method of claim 91 wherein said adding properties is
2 performed by a computer in conjunction with a user.

1 99. The method of claim 98 wherein said adding properties prompts
2 a user to add a property to the ontology model when there is an XML element or
3 an XML attribute described in the at least one XML schema that does not
4 correspond to an existing property or composition of properties in the ontology
5 model.

file
1 1.126

2 ¶100. The method of claim 91 wherein said adding properties is
performed automatically by a computer.

1 101. The method of claim 100 wherein said adding properties
2 automatically adds a property to the ontology model when there is an XML
3 element or an XML attribute described in the at least one relational database
4 schema that does not correspond to an existing property or composition of
5 properties in the ontology model.

1 102. A system for building an ontology model into which data schema
2 can be embedded, comprising:
3 a schema receiver receiving at least one data schema; and
4 a model builder building an ontology model into which the at
5 least one data schema can be embedded.

1 103. The system of claim 102 further comprising a schema format
2 convertor, converting at least one of the at least one data schema from an external
3 format to an internal format.

1 104. The system of claim 102 wherein the at least one data schema is
2 at least one table schema describing data tables having columns.

1 105. The system of claim 104 wherein the at least one table schema is
2 at least one relational database schema describing relational database tables.

1 106. The system of claim 105 further comprising an ontology receiver
2 receiving an initial ontology model, and wherein said model builder comprises:
3 a class adder adding classes to the initial ontology model
4 corresponding to tables described in the at least one relational database schema;
5 and

6 a property adder adding properties to the initial ontology model
7 corresponding to table columns described in the at least one relational database
8 schema.

1 107. The system of claim 106 wherein the initial ontology model is
2 empty.

1 108. The system of claim 106 wherein the initial ontology model is
2 non-empty.

1 109. The system of claim 108 further comprising an ontology format
2 convertor, converting the initial ontology model from an external format to an
3 internal format.

1 110. The system of claim 106 wherein said class adder is guided by a
2 user in conjunction with a computer.

1 111. The system of claim 110 wherein said class adder prompts a user
2 to add a class to the ontology model when there is a table described in the at least
3 one relational database schema that does not correspond to an existing class in the
4 ontology model.

1 112. The system of claim 106 wherein said class adder is
2 automatically guided by a computer.

1 113. The system of claim 112 wherein said class adder automatically
2 adds a class to the ontology model when there is a table described in the at least
3 one relational database schema that does not correspond to an existing class in the
4 ontology model.

1 114. The system of claim 106 wherein said property adder is guided
2 by a user in conjunction with a computer.

1 115. The system of claim 114 wherein said property adder prompts a
2 user to add a property to the ontology model when there is a table column
3 described in the at least one relational database schema that does not correspond
4 to an existing property or composition of properties in the ontology model.

1 116. The system of claim 106 wherein said property adder is
2 automatically guided by a computer.

1 117. The system of claim 116 wherein said property adder
2 automatically adds a property to the ontology model when there is a table column
3 described in the at least one relational database schema that does not correspond
4 to an existing property or composition of properties in the ontology model.

1 118. The system of claim 105 wherein said model builder comprises
2 an inheritance processor inferring inheritance relationships between classes in the
3 ontology model based on relationships between tables in the at least one relational
4 database schema.

1 119. The system of claim 118 wherein said inheritance processor
2 infers that a first class in the ontology model inherits from a second class in the
3 ontology model when a table corresponding to the first class has a primary key
4 that is a foreign key to a table corresponding to the second class.

1 120. The system of claim 118 wherein said model builder ensures that
2 classes corresponding to tables in the at least one relational database schema obey
3 the inferred inheritance relationships.

1 121. The system of claim 120 wherein said inheritance processor
2 prompts a user to confirm an inferred inheritance relationship.

1 122. The system of claim 102 wherein the at least one data schema
2 comprises at least one document schema describing documents.

1 123. The system of claim 122 wherein the at least one document
2 schema comprises at least one XML schema that describes XML documents,
3 wherein having at least one XML complexType with at least one XML element or
4 XML attribute.

1 124. The system of claim 123 further comprising an ontology receiver
2 receiving an initial ontology model, and wherein said model builder comprises:

3 a class adder adding classes to the initial ontology model
4 corresponding to XML complexTypes described in the at least one XML schema;
5 and

6 a property adder adding properties to the initial ontology model
7 corresponding to table columns in the at least one relational database schema.

1 125. The system of claim 124 wherein the initial ontology model is
2 empty.

1 126. The system of claim 124 wherein the initial ontology model is
2 non-empty.

1 127. The system of claim 124 wherein said class adder is guided by a
2 user in conjunction with a computer.

1 128. The system of claim 127 wherein said class adder prompts a user
2 to add a class to the ontology model when there is an XML complexType
3 described in the at least one XML schema that does not correspond to an existing
4 class in the ontology model.

1 129. The system of claim 124 wherein said class adder is
2 automatically guided by a computer.

1 130. The system of claim 129 wherein said class adder automatically
2 adds a class to the ontology model when there is an XML complexType described
3 in the at least one XML schema that does not correspond to an existing class in
4 the ontology model.

1 131. The system of claim 124 wherein said property adder is guided
2 by a user in conjunction with a computer.

1 132. The system of claim 131 wherein said property adder prompts a
2 user to add a property to the ontology model when there is an XML element or
3 XML attribute described in the at least one XML schema that does not correspond
4 to an existing property or composition of properties in the ontology model.

1 133. The system of claim 124 wherein said property adder is
2 automatically guided by a computer.

1 134. The system of claim 133 wherein said property adder
2 automatically adds a property to the ontology model when there is an XML
3 element or XML attribute described in the at least one XML schema that does not
4 correspond to an existing property or composition of properties in the ontology
5 model.

1 135. An article of manufacture including one or more computer-
2 readable media that embody a program of instructions for transforming data from
3 one schema to another, wherein the program of instructions, when executed by a
4 processing system, causes the processing system to:
5 receive a source data schema and a target data schema;
6 map the source data schema into an ontology model;
7 map the target data schema into the ontology model; and
8 derive a transformation for transforming data conforming to the
9 source data schema into data conforming to the target relational database schema,
10 using the ontology model.

1 136. The article of claim 135 wherein the one or more computer-
2 readable media include one or more non-volatile storage devices.

1 137. The article of claim 135 wherein the one or more compute-
2 readable media include a carrier wave modulated with a data signal.

1 138. An article of manufacture including one or more computer-
2 readable media that embody a program of instructions for building a common
3 ontology model into which data schema can be embedded, wherein the program
4 of instructions, when executed by a processing system, causes the processing
5 system to:
6 receive at least one data schema; and
7 build an ontology model into which the at least one data schema
8 can be embedded.

1 139. The article of claim 138 wherein the one or more computer-
2 readable media include one or more non-volatile storage devices.

1 140. The article of claim 138 wherein the one or more compute-
2 readable media include a carrier wave modulated with a data signal.

10052045-01150